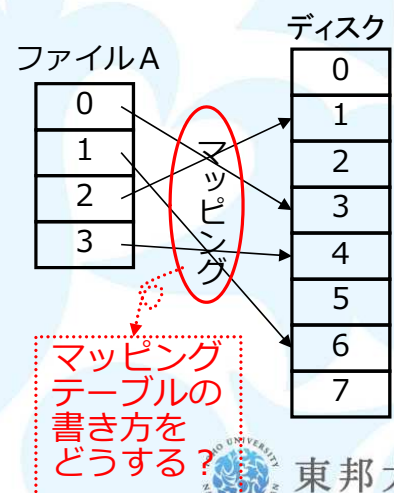
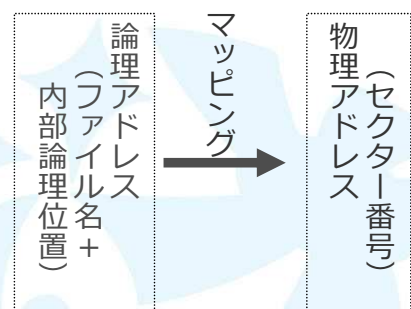


ファイルの 記憶空間管理 FAT

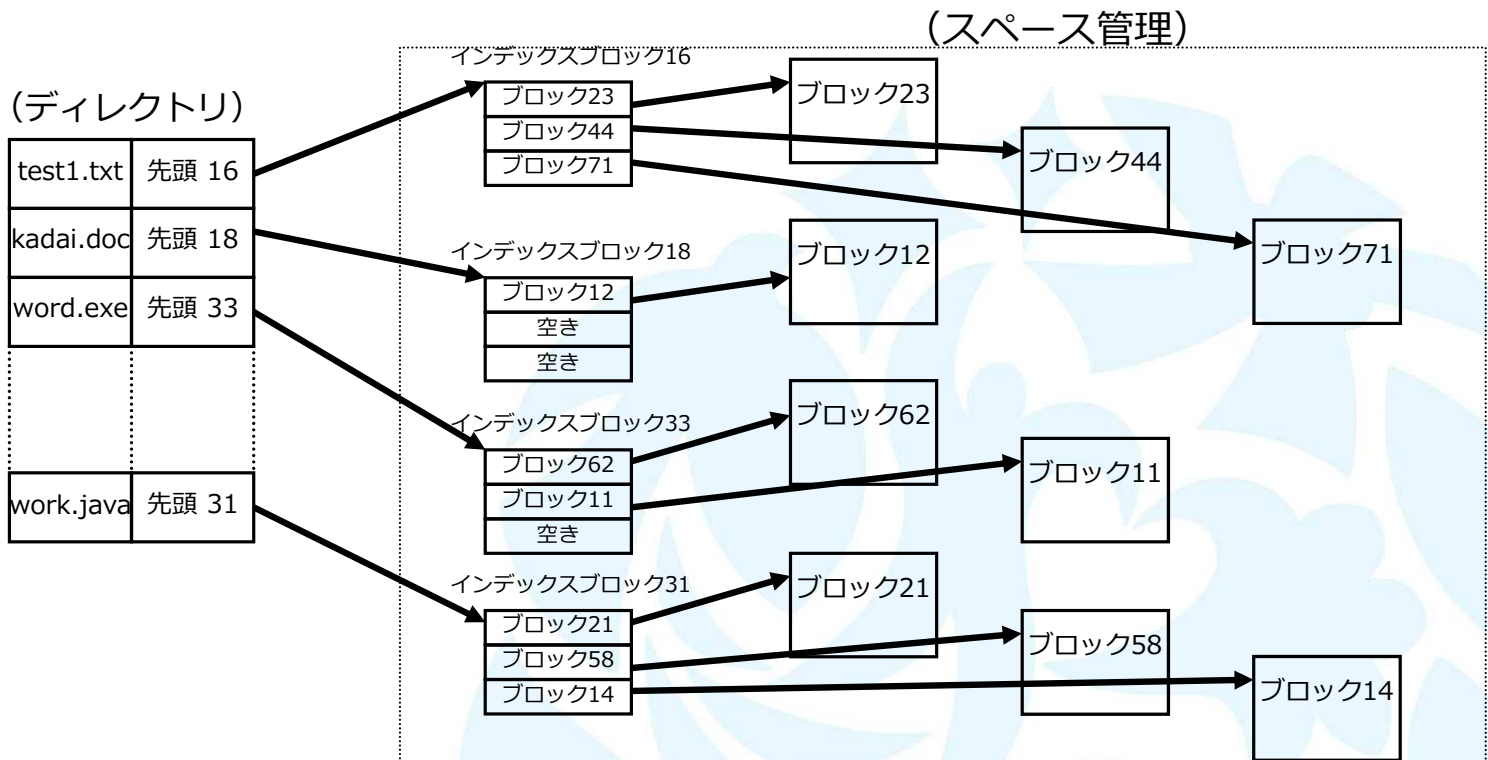


管理の考え方

- ブロックの「論理アドレス」と物理アドレス間をマッピング
- ブロックの順番が変わっても（途中追加など） OK
- 縮んでも隙間が（断片化）出来ない
- 仮想記憶と似たような発想
 - アドレスをブロック単位でマップ
⇒上記の問題解決

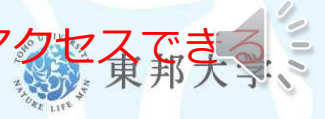


スペース管理用の情報を集める



- **ブロック71へ至るのも早い**
インデックスブロック16を1つ読めば23も44も71もアクセスできる

2



実用されている仕組の例

- FAT File Allocation Table
 - MS DOS・Windows（互換のため）で使われている
 - 元々フロッピーディスク、8-16ビットCPU前提

3



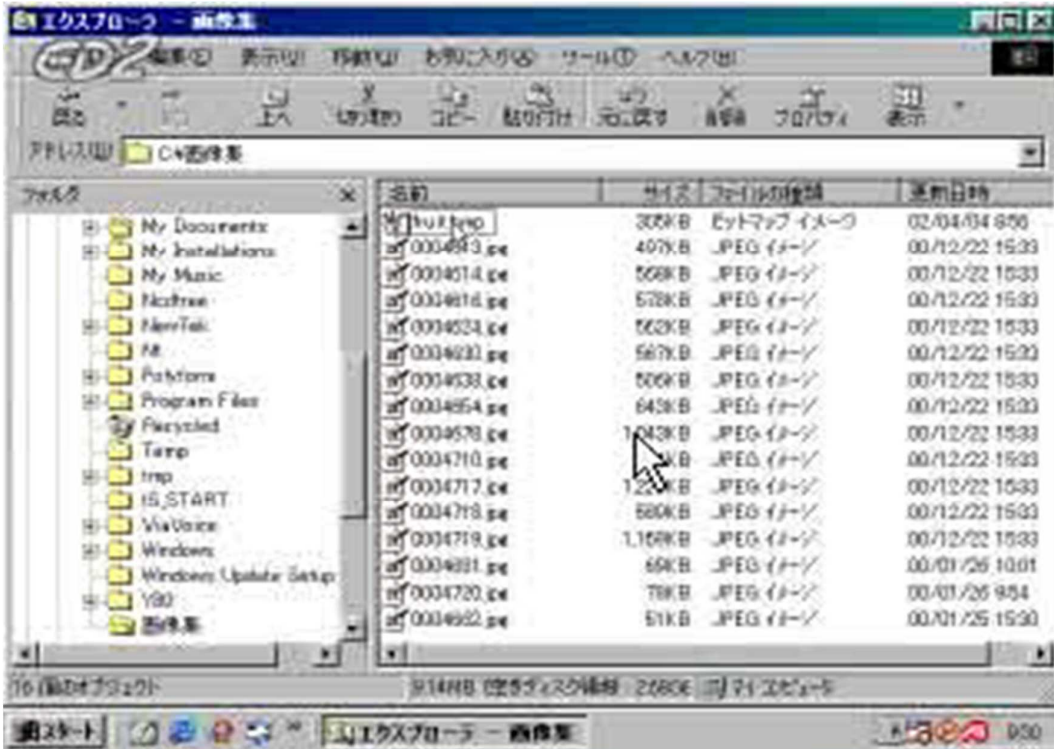
実用されている仕組の例

- FAT File Allocation Table
 - MS DOS・Windows（互換のため）で使われている
 - 元々フロッピーディスク、8-16ビットCPU前提
- UNIXのファイルシステム（UFS、i-node、ext2）
 - 多段のインデックスブロックの工夫

実用されている仕組の例

- FAT File Allocation Table
 - MS DOS・Windows（互換のため）で使われている
 - 元々フロッピーディスク、8-16ビットCPU前提
- UNIXのファイルシステム（UFS、i-node、ext2）
 - 多段のインデックスブロックの工夫
- もっと新しい仕組が出ている
 - MS Win XP/VistaではNTFS
 - Linuxではext3, ReiserFS, JFS/XFS

実用されている仕組み① F A T



- <http://www.sugilab.net/jk/joho-kiki/1505/index.html>

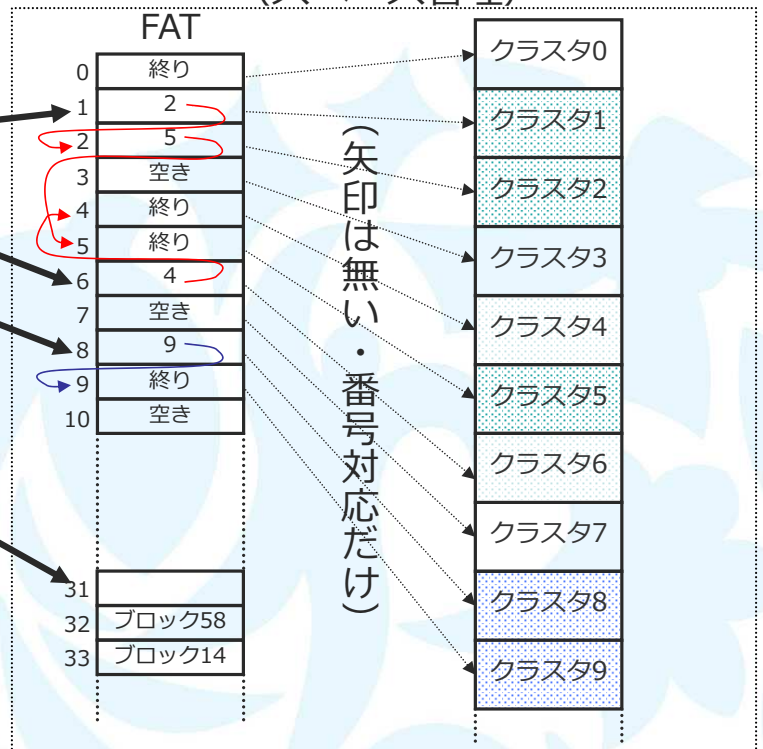


実用されている仕組み① F A T

(スペース管理)

(ディレクトリ)

test1.txt	先頭 1
kadai.doc	先頭 6
word.exe	先頭 8
...	...
work.java	先頭 31



- FATテーブルにリンクが書かれている

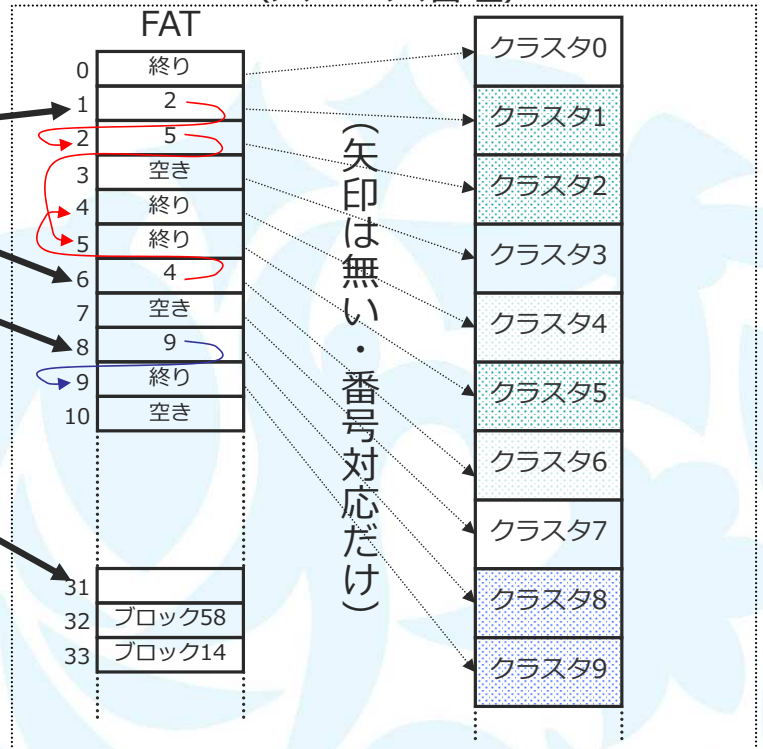


実用されている仕組み① F A T

(スペース管理)

(ディレクトリ)

test1.txt	先頭 1
kadai.doc	先頭 6
word.exe	先頭 8
work.java	先頭 31



- FATテーブルの中で、クラスタ間のつながりを記述する

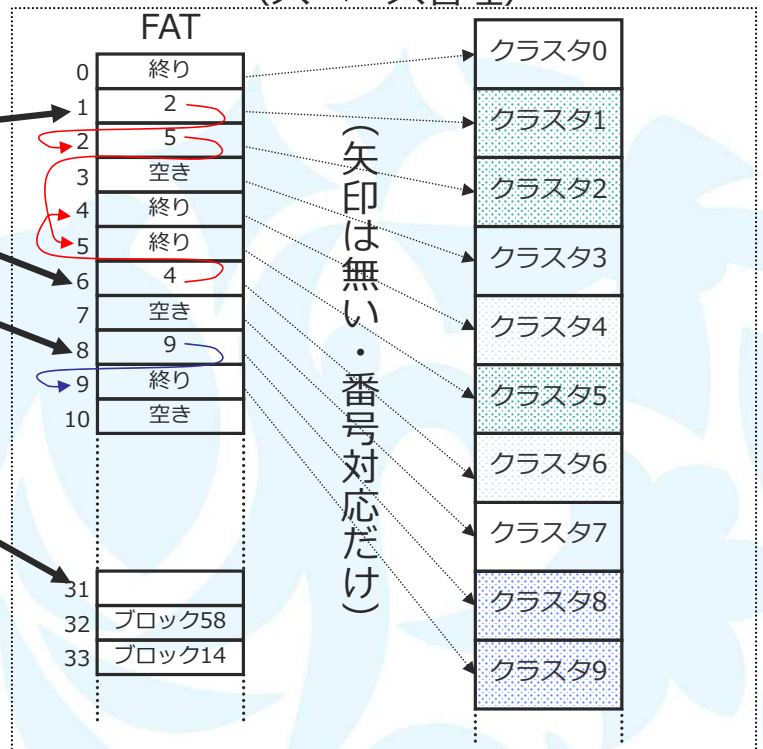


実用されている仕組み① F A T

(スペース管理)

(ディレクトリ)

test1.txt	先頭 1
kadai.doc	先頭 6
word.exe	先頭 8
work.java	先頭 31



- FATテーブルの中で、クラスタ間のつながりを記述する
- チェーン状だが、途中要素は同じFATブロックなのでディスクを毎回読まなくて済む



実用されている仕組み① F A T

- クラスタ

- 1ブロックではなく、複数の物理ブロック（セクタ）を集めてFATテーブルの管理の単位とした

実用されている仕組み① F A T

- クラスタ

- 1ブロックではなく、複数の物理ブロック（セクタ）を集めてFATテーブルの管理の単位とした
- （注）ディスク全体をクラスタに分割するのでディスク容量が増えるとクラスタサイズを大きくせざるを得なかった
大きくすると、スペースの無駄が出る

実用されている仕組み① F A T

- クラスタ

- 1ブロックではなく、複数の物理ブロック（セクタ）を集めてFATテーブルの管理の単位とした
- （注）ディスク全体をクラスタに分割するのでディスク容量が増えるとクラスタサイズを大きくせざるを得なかった
大きくすると、スペースの無駄が出る

- FAT12, FAT16, FAT32

- ディスクの容量増大に伴って、クラスタ番号の桁数が不足し、ビット数を増やした

最初はフロッピーディスク（2HDだと1.44MB）だったが
その後ハードディスクでも使われた

12



F A T の問題

- ディスク容量の増加に対応しづらい（前頁）

13



F A T の問題

- ディスク容量の増加に対応しづらい（前頁）
- FATテーブルを大きくすると保管が大変
 - FATはクラスタ個数分のエントリ⇒大きくなる

14

F A T の問題

- ディスク容量の増加に対応しづらい（前頁）
- FATテーブルを大きくすると保管が大変
 - FATはクラスタ個数分のエントリ⇒大きくなる
- FATテーブルの破壊に弱い
 - FATが読めなければファイルは読めない
 - 一応二重化してある（2つ同じFATを持つ）がよく壊れてディスク全体が読めなくなった

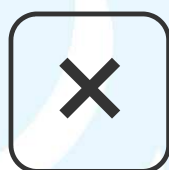
15

FATの問題

- ディスク容量の増加に対応しづらい（前頁）
- FATテーブルを大きくすると保管が大変
 - FATはクラスタ個数分のエントリ⇒大きくなる
- FATテーブルの破壊に弱い
 - FATが読めなければファイルは読めない
 - 一応二重化してある（2つ同じFATを持つ）がよく壊れてディスク全体が読めなくなった
- これらのことから、Win XP・VistaではNTFS（ジャーナル型ファイルシステム）を推奨

16

FATによる
記憶空間管理の仕組が
理解できましたか？



↓
次へ

17